What is claimed is:

1. A pipelined microprocessor for processing instructions, comprising at least one pipeline comprising an instruction fetching functional stage, an instruction decoding functional stage, an execution functional stage comprising a number of execution units, and a commit functional stage comprising or being associated with a reorder buffer, wherein the microprocessor further includes detecting means for detecting instruction irregularities, such that when an instruction irregularity is detected, an irregularity indication and a flush instruction are generated, and the irregularity indication is used to initiate a flush mode, whereas the flush instruction, when received in a functional stage or unit set in flush mode, resets the flush mode in said functional stage/unit.

2. The pipelined microprocessor according to claim 1, wherein the detecting instruction irregularities at the execution functional stage.

3. The pipelined microprocessor according to claim 1, wherein the detecting means are provided for detecting instruction irregularities at the commit functional stage.

4. The pipelined microprocessor according to claim 1, wherein the irregularity indication at least initiates the flush mode in the execution unit of the execution functional stage handling the instruction irregularity.

5. The pipelined microprocessor according to claim 3, wherein the irregularity indication at least initiates the flush mode in the commit functional stage.

6. The pipelined microprocessor according to claim 5, wherein the execution units of the execution functional stage are set in flush mode by the irregularity indication.

7. The pipelined microprocessor according to claim 4, wherein the irregularity indication is also provided to the decoding functional stage to set the decoding functional stage in flush mode.

8. The pipelined microprocessor according to claim 1, wherein the instruction irregularity is provided with an irregularity marking.

9. The pipelined microprocessor according to claim 1, wherein the instruction irregularity is provided with an irregularity marking and the commit functional stage is set in flush mode upon reception/committing of the instruction irregularity when not already in flush mode.

10. The pipeline microprocessor according to claim 1, wherein all instructions processed by a unit/functional stage in flush mode are provided with a cancel marking, unless already marked, and said cancel marking is used to indicate to units/functional stages, in front of the unit/functional stage that the flush mode should be set.

11. The pipelined microprocessor according to claim 1, wherein the flush instruction is processed through at least some of the functional stages of the pipeline.

12. The pipelined microprocessor according to claim 1, wherein at least branch instructions are checked to establish if they comprise an instruction irregularity in the same order as they where provided to the decoding functional stage.

13. The pipelined microprocessor according to claim 1, wherein a reservation functional stage comprising a number of reservation units is provided preceding the execution functional stage and at least some of the reservation units of the reservation functional stage process(es)/arrange(s) instructions in order.

14. The pipelined microprocessor according to claim 1, wherein no command is issued to external units from an execution unit handling an instruction irregularity during the flush mode.

15. The pipelined microprocessor according to claim 1, wherein the pipelined microprocessor is adapted to handle instruction irregularities comprising exceptions and/or interrupts (mis)predicted branch instructions.

16. The pipelined microprocessor according to claim 15, wherein the pipelined microprocessor is adapted to operate as a speculative out-of-order processor that predicts whether a conditional branch is to be taken or not.

17. The pipelined microprocessor according to claim 13, wherein branch instructions are provided in order in a reservation unit communicating with a execution unit handling branch instructions.

18. The pipelined microprocessor according to claim 16, wherein a conditional branch instruction is predicted at the instruction fetching functional stage, information about the outcome is provided from the execution unit handling the predicted branch instruction to the detecting means for establishing whether the prediction was correct or not, and a response is returned to said execution unit if the prediction was correct.

19. The pipelined microprocessor according to claim 18, wherein if it is established that a prediction was incorrect, a response is provided to the execution unit, the response providing for the generation of the irregularity indication (signal) from the detecting means.

20. The pipelined microprocessor according to claim 19, wherein the flush instruction is generated in the execution unit.

21. The pipelined microprocessor according to claim 19, wherein the flush instruction is generated at the decoding functional stage, at the instruction fetching functional stage, or externally of the pipeline.

22. The pipelined microprocessor according to claim 1, wherein the detecting means operate in association with the execution unit or the commit functional stage.

23. The pipelined microprocessor according to claim 1, wherein the detecting means is provided separately or externally of the pipeline.

24. The pipelined microprocessor according to claim 23, wherein when the flush instruction is generated, an irregularity indication is sent to the execution functional stage indicating the provision of the flush instruction and to initiate the flush mode in the execution unit handling the instruction irregularity.

25. The pipelined microprocessor according to claim 17, wherein the detecting means establishing the correctness of a prediction generates the flush instruction if the prediction was incorrect.

26. The pipelined microprocessor according to claim 17, wherein the detecting means are comprised in the instruction fetching functional stage.

27. The pipelined microprocessor according to claim 17, wherein the detecting means are comprised by instruction handling means translating macrocode or assembler code to microcode or microcode to microcode for input to the instruction fetching functional stage.

28. The pipelined microprocessor according to claim 27, wherein additional flush instruction detection means are associated with the instruction fetching functional stage or the instruction handling means for detecting the generation of a flush instruction or a misprediction and sending the irregularity indication to the execution unit handling the predicted branch instruction.

29. The pipelined microprocessor according to claim 1, wherein during flush mode, all passing instructions are ignored until the arrival of the flush instruction resetting the flush mode.

30. A method of processing instructions in a pipelined microprocessor comprising at least one pipeline comprising an instruction fetching functional stage, an instruction decoding functional stage, an execution functional stage and a commit functional stage, the method comprising the steps of:

detecting an instruction irregularity at the execution functional stage or at the commit functional stage;

providing an irregularity indication to set at least the execution unit handling the instruction irregularity or the commit functional stage in flush mode;

generating a flush instruction;

inputting the flush instruction to the pipeline;

cancel marking instructions processed in a unit/functional stage in flush mode; and

resetting the flush mode in each unit/functional stage in flush mode upon reception of the flush instruction.

31. The method according to claim 30, further comprising the step of:

marking the instruction causing the irregularity when detecting the instruction irregularity.

32. The method according to claim 31, further comprising the step of:

setting the decode functional stage in flush mode through initiation by said irregularity indication.

33. The method according to claim 32, further comprising the step of:

setting the commit functional stage in flush mode, wherein the flush mode in the commit functional stage is initiated either by the irregularity indication or when committing the irregularity marked instruction.

34. The method according to claim 30, wherein the step for detecting an instruction irregularity comprises:

communicating information about the instruction to the instruction fetching functional stage keeping actual address information about the instruction;

verifying the correctness of the outcome of the instruction;

if no instruction irregularity is established, then sending a response to the execution unit; and

if the instruction irregularity is true, detecting that no response is received after a given time interval, making the execution unit aware of the instruction irregularity, and generating a

flush instruction.

35. The method according to claim 30, wherein the instruction irregularity is a mispredicted branch prediction and the detection step comprises:

providing information to the instruction fetching functional stage about the outcome of the branch prediction including an address of the predicted branch target;

comparing the predicted branch target address with the actual branch target address; and

if they are not the same, generating a flush instruction.

36. The method according to claim 30, wherein the step of detecting an instruction irregularity comprises:

predicting if an instruction might cause an irregularity;

attaching the result of the predicted outcome to said instruction;

processing the instruction in an execution unit and evaluating if the instruction caused an irregularity;

detecting if the prediction was correct or not;

generating an irregularity indication in the execution unit handling the instruction irregularity or in the commit functional stage, if the prediction was incorrect; and

proceeding with unaffected processing when the prediction is correct.

37. The method according to claim 30, comprising the additional step of:

providing instructions, in order, in at least some reservation unit(s) that to precede at least an execution unit handling branch instructions.